PYTHON FOR OSINT

AUTOMATING YOUR OSINT COLLECTION USING PYTHON LIBRARIES

BY EMILY CHANCE AND AYELET GLASER

ABOUT EMILY

- OSINT consultant in the Los Angeles Area
- Specializes in doing threat assessments and other OSINT gathering for SMBs and private clients
- Self-taught python developer
- Defcon562 member
- Can be found on Twitter at @g_solaria
- Probably a bit too obsessed with dogs

ABOUT AYELET GLASER

- Based in France
- Freelance python and database dev
- Available for freelance work (contact Emily for info)
- Conversely, obsessive cat mom to Squeaker and Potato

PURPOSE OF THIS TALK

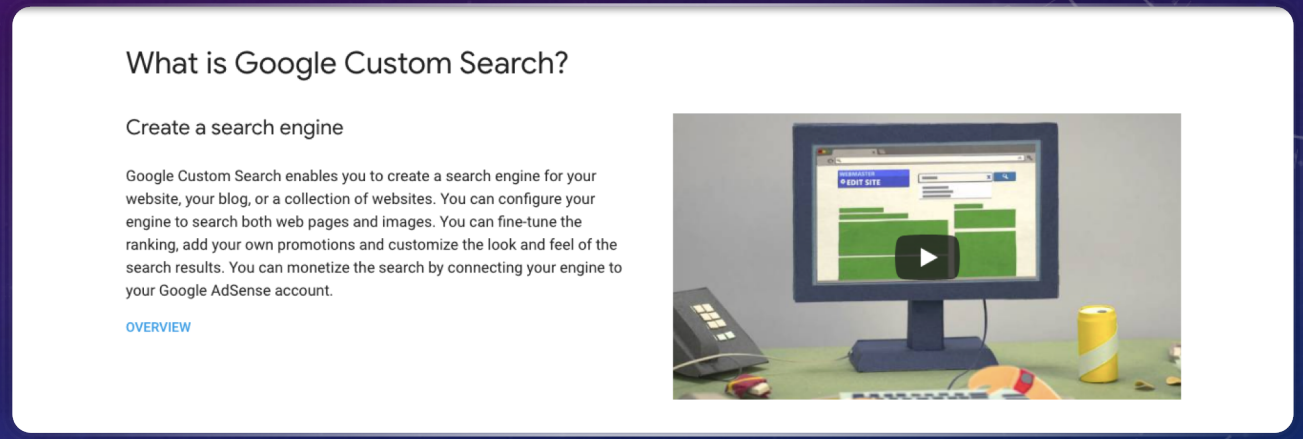
- This talk will cover several python libraries that can be used to automate Open Source Intelligence (OSINT) gathering.
- This talk assumes that you already are familiar with OSINT, and that you have some experience writing python.
- We will walk through the libraries and show examples of programs that use them.
- Programs can be found online at <insert link>
- Programs are written in python 3 like the good lord intended.
- Some code examples may not run on their own because they are meant to demonstrate one specific concept.

LIBRARIES COVERED

- **Google Custom Search API** (https://developers.google.com/custom-search/)
- **Beautiful Soup** (https://www.crummy.com/software/BeautifulSoup/)
- **Google Cloud Language API** (https://cloud.google.com/natural-language/)
- **Fuzzywuzzy** (https://github.com/seatgeek/fuzzywuzzy)
- **Selenium** (https://www.seleniumhq.org/)

GOOGLE CUSTOM SEARCH API

- Google provides the ability to create custom search engines that hit specific domains.
- This is restrictive in that we can only look at the domains we've specified when creating the CSE.
- Workaround – go back in and remove those domains, and we now have an API for all of Google.



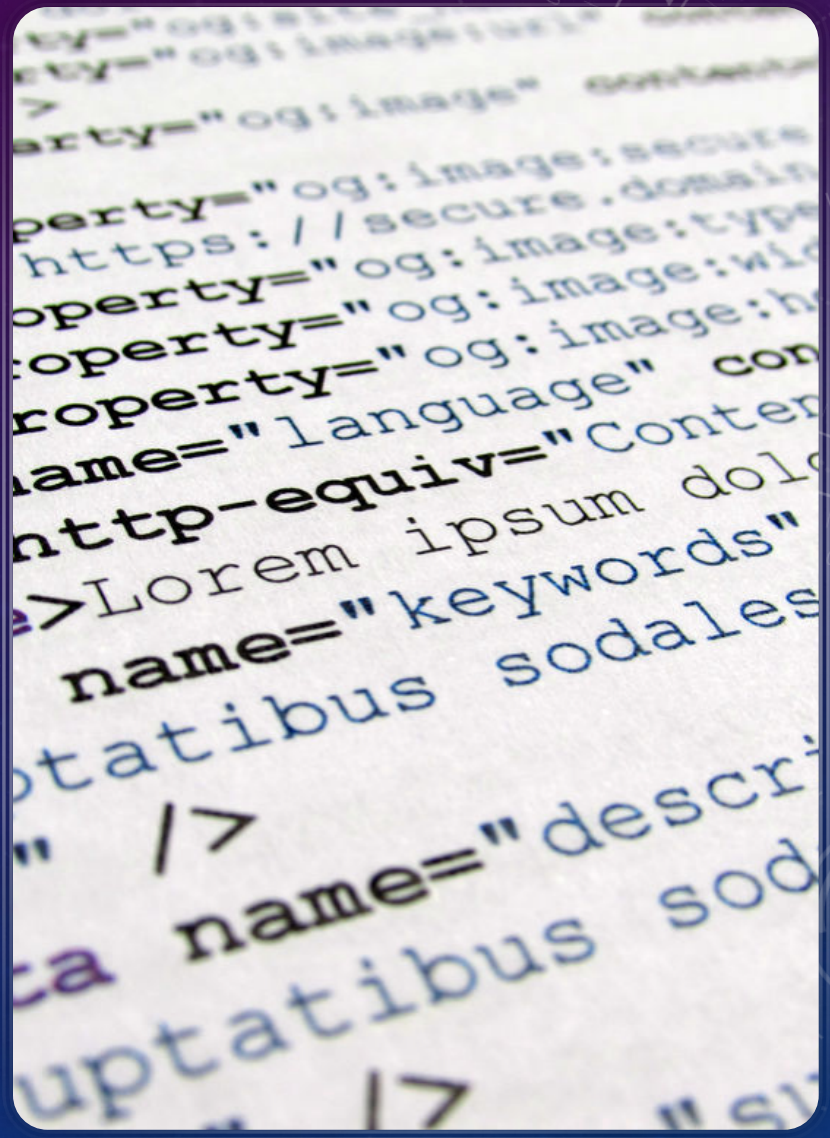
CREATING A CSE

- To create a new Google Custom Search Engine, go to https://cse.google.com/cse/all
- Click “add” and fill out the information.
  - Note that it will ask you specify a domain you want to search. We will delete this later.
- Click Create, then on the next page, click Control Panel.
- Select the domain you added, and click delete. You now have an API for Google.
- Up to 100 searches per day for free. Searches are $5 per 1000 after that, up to 10,000/day.
- Creating a new Google Cloud billing account gives you $300 in credit – you’ll just need to change your api key and cse ID.
- Install library using pip:
  - `pip install --upgrade google-api-python-client`

```python
from googleapiclient.discovery import build
from makeKeywordList import makeKeywordList
from makeWebsiteList import makeWebsiteList
import json

websiteList = makeWebsiteList()
keywordList = makeKeywordList()

# Google CSE api key and search engine ID.
keyFile = '.monitor.api'
engineFile = '.search.id'

# read API key and CSE ID into memory
with open(keyFile) as f_obj:
    my_api_key = f_obj.read().rstrip()

with open(engineFile) as f_obj:
    my_cse_id = f_obj.read().rstrip()

queryList = []
for website in websiteList:
    query = "site:" + website.rstrip() + ' ' + '"' + keywordList[0].rstrip() + '"'
    queryList.append(query)

def google_search(search_term, api_key, cse_id, **kwargs):
        service = build("customsearch", "v1", developerKey=api_key)
        res = service.cse().list(q=search_term, cx=cse_id, **kwargs).execute()
        try:
                return res['items']
        except KeyError:
                pass

resultsList = []

for query in queryList:
    results = google_search(query, my_api_key, my_cse_id, num=10)
    try:
        for result in results:
            resultsList.append(result['link'])
    except TypeError:
        pass

for r in resultsList:
        print(r)
```

BEAUTIFUL SOUP

- HTML parsing library.
- Creates a “Beautiful Soup” object that maintains the structure of the page and can be worked with in multiple ways.
- Allows you to select portions of text from a page based on tags, classes, etc.
- Allows you to find() or find_all() for locating specific words or phrases within sections of the page.
- Install using pip:
  - `pip install bs4`



```
import bs4, re, requests, time, timeout_decorator

with open('keywordfile.txt') as keywords:
        keywordList = keywords.readlines()

@timeout_decorator.timeout(15, timeout_exception=StopIteration)
def priorityFirst(htmlSoup):
    """Counts instances of the first keyword"""
    search_word = keywordList[0].rstrip()
    keys = htmlSoup.body.find_all(strings=re.compile('.*{0}*'.format(search_word)), recursive=True)
    return keys

@timeout_decorator.timeout(15, timeout_exception=StopIteration)
def prioritySecond(htmlSoup):
    """Counts instances of the second keyword"""
    search_word = keywordList[1].rstrip()
    keys = htmlSoup.body.find_all(strings=re.compile('.*{0}*'.format(search_word)), recursive =True)
    return keys

@timeout_decorator.timeout(15, timeout_exception=StopIteration)
def priorityRest(htmlSoup):
    """Counts instances of the rest of the keywords"""
    secondKeylist = keywordList[2:-1]
    for keyword in secondKeylist:
        search_word = keyword
        keys = htmlSoup.body.find_all(string=re.compile('.*{0}*'.format(search_word)), recursive=True)
        return keys

with open('output.txt') as output:
        resultList = output.readlines()

```

GOOGLE CLOUD LANGUAGE API

- Can be created using the same account as the CSE API.
- Installed via pip
  - `pip install google-cloud-language`
- Gives you a JSON file containing your credentials, which must be exported to your environment
  - `export GOOGLE_APPLICATION_CREDENTIALS='credential-doc.json'`
- Allows for analysis of documents to find entities, sentiment, and sentiment about entities.
- There is a cost to the API calls, but as with the CSE api, creating new accounts will get you $300 credit.

```python
from google.cloud import language
import requests
import bs4
import time
import timeout_decorator

with open('output.txt', 'r') as f_obj:
        resultsList = f_obj.readlines()

#for r in resultsList:
#       print(r)


def get_sentiment(page):

        client = language.LanguageServiceClient()

        document = language.types.Document(content=page, type="PLAIN_TEXT")

        response = client.analyze_sentiment(document=document, encoding_type="UTF32")

        sentiment = response.document_sentiment

        print(sentiment.score)
        print(sentiment.magnitude)

for r in resultsList:
        page = requests.get(r)

        try:
                html = page.text

                soup = bs4.BeautifulSoup(html, "lxml")

                body = soup.body.get_text()

                print(r)

                sentiment = get_sentiment(body)


        except AttributeError:
                pass
        except Exception as e:
                print("There was an exception: %s" % (e))
```

FUZZYWUZZY

- Uses to compare for similarities between words or phrases.
- Can be used to identify keywords in a page that may have been misspelled, etc.
- Uses **Levenshtein distance** to compare words or phrases.
  - **Levenshtein distance** is a string metric for measuring the difference between two sequences. Informally, the Levenshtein distance between two words is the minimum number of single-character edits (insertions, deletions or substitutions) required to change one word into the other.
- Installed via pip:
  - `pip install fuzzywuzzy[speedup]`

```python
from fuzzywuzzy import fuzz
import bs4
from google.cloud import language
import requests

with open('keywordfile.txt') as f_obj:
        keywords = f_obj.readlines()

with open('output.txt') as f_obj:
        results = f_obj.readlines()

def get_ents(page):

        ents = []

        client = language.LanguageServiceClient()

        document = language.types.Document(content=page, type="HTML")

        response = client.analyze_entities(document=document, encoding_type="UTF32")

        for ent in response.entities:
                ents.append(ent.name)

        return ents

def match(ents):

        matchedEnts = []

        for ent in ents:
                score = fuzz.partial_ratio(ent, keywords[0])
                if score > 90:
                        matchedEnts.append(ent)

        return matchedEnts

for r in results:

        try:
                print("Matching...")
                page = requests.get(r)

                bsObj = bs4.BeautifulSoup(page.text, "lxml").get_text()

                ents = get_ents(bsObj)

                matchedEnts = match(ents)

        except Exception as e:
                print(e)
```

SELENIUM

- Emulates a web browser.
- Can be used to do things like log into websites.
- Allows you to use different browsers, including headless chrome.
- Installed via pip:
  - `pip install selenium.`
- Can be passed credentials to log into websites, handle cookies, and take screenshots.

```python
from selenium import webdriver

profileId = input("Please enter a Facebook UserID: ")

with open('fbpagelist.txt') as pl:
        urlEnds = pl.readlines()

urls = []

for ue in urlEnds:
        url = "https://facebook.com" + profileId + ue
        urls.append(url)

def browser(url):

        options = webdriver.ChromeOptions()

        options.binary_location = '/Applications/Google Chrome Canary.app/Contents/MacOS/Google Chrome Canary'

        options.add_argument('headless')

        driver = webdriver.Chrome(chrome_options=options)

        driver.get('https://facebook.com')

        driver.implicitly_wait(10)

        email = driver.find_element_by_css_selector('input[type=email]')
        password = driver.find_element_by_css_selector('input[type=password]')
        login = driver.find_element_by_css_selector('input[value="Log In"]')

        email.send_keys('hallieamberhall@gmail.com')
        password.send_keys('5`X7b2q&Ef=U~Z`V')

        driver.get_screenshot_as_file(ue + 'page.png')

for url in urls:
        browser(url)
```

XONITOR

- Xonitor (re-written from the original by Ayelet) combines elements of the first four libraries (Selenium support for logging in to sites and screenshotting results to be added).
- Allows input of keywords and websites to search.
- Uses CSE to find results, Cloud Language API get entity sentiments, and keyword weighting to come up with a priority score.
- Can be used to monitor results and prioritize them based on keyword concentration and the various entities.
- Run as cron job to monitor results consistently.
- Open Sourced at (repo url)

keys.txt

```
1 Jeff Bezos
2 Amazon
3 Amazon Employees
4 Union
5 Unionize
```

sites.txt

```
1 washingtonpost.com
2 linkedin.com
3 bloomberg.com
4 nytimes.com
5 facebook.com
6 seattletimes.com
7 seattlepi.com
8 thestranger.com
9 seattleweekly.com
```

URL	Priority
https://m.seattlepi.com/technology/businessinsider/article/AMAZON-CEO-I-Want-To-See-Millions-Of-People-5495986.php	620
https://www.nytimes.com/2018/01/12/technology/jeff-bezos-amazon.html	380
https://www.linkedin.com/showcase/jeff-bezos-newslines	310
https://www.seattlepi.com/business/amp/Jeff-Bezos-is-planning-his-big-philanthropic-12997579.php	290
https://www.washingtonpost.com/news/comic-riffs/wp/2016/07/16/from-boyhood-to-hollywood-jeff-bezos-who-played-star-trek-as-a-kid-has-a-role-in-star-trek-beyonc	260
https://www.linkedin.com/pulse/customer-support-done-right-jeff-bezos-e-mail-floors-indian-p-k	220
https://www.linkedin.com/pulse/use-5-whys-find-root-causes-peter-abilla	220
https://www.nytimes.com/2018/09/13/technology/jeff-and-mackenzie-bezos-pledge-2-billion-for-homeless-and-preschoolers.html	220
https://www.seattletimes.com/business/amazon/jeff-bezos-launches-2-billion-philanthropic-fund-targeting-homelessness-education-in-low-income-communities/	220
https://www.seattletimes.com/business/amazon/amazons-jeff-bezos-chimes-in-on-10000-year-clock-as-installation-starts/	220
https://www.seattletimes.com/business/amazon/the-seattle-times-first-story-about-jeff-bezos-when-you-had-to-hook-up-to-the-internet-22-years-ago/	220
https://www.linkedin.com/pulse/20141201155043-461078-jeff-bezos-a-profile-in-failure	210
https://www.linkedin.com/pulse/does-your-ceo-read-customer-complaints-amazons-jeff-bezos-schaffer	210
https://www.seattlepi.com/business/article/Rose-McGowan-goes-after-Amazon-and-Jeff-Bezos-in-12273993.php	210
https://www.washingtonpost.com/education/2018/09/19/jeff-bezos-is-spending-billion-help-homeless-educate-poor-kids-sounds-good-is-it/	170
https://www.nytimes.com/2018/09/05/technology/jeff-bezos-amazon-political-donation-veterans.html	170
https://www.nytimes.com/2017/07/27/business/jeff-bezos-richest-man.html	170
https://www.seattlepi.com/business/techwire/article/Bernie-Sanders-slams-Amazon-and-Jeff-Bezos-rich-12941090.php	170
https://www.seattlepi.com/business/article/Silence-is-the-way-Amazon-meetings-start-12878427.php	170
https://www.thestranger.com/slog/2018/06/12/27505243/jeff-bezos-owns-seattle-direct-your-complaints-to-him	170
https://www.thestranger.com/slog/2018/08/10/30563251/jeff-bezos-dreams-of-shooting-his-payload-into-space-while-we-all-pay-for-amazons-carbon-footprint	170
https://www.thestranger.com/slog/2018/07/16/29260492/bezos-has-150-billion-saves-hundreds-of-millions-in-washington-income-tax	170
https://www.washingtonpost.com/national/washington-post-to-be-sold-to-jeff-bezos/2013/08/05/ca537c9e-fe0c-11e2-9711-3708310f6f4d_story.html	130
https://www.washingtonpost.com/business/economy/washington-post-closes-sale-to-amazon-founder-jeff-bezos/2013/10/01/fca3b16a-2acf-11e3-97a3-ff2758228523_	130
https://www.washingtonpost.com/national/jeff-bezos-on-post-purchase/2013/08/05/e5b293de-fe0d-11e2-9711-3708310f6f4d_story.html	130
https://www.washingtonpost.com/news/reliable-source/wp/2017/01/12/jeff-bezos-is-the-anonymous-buyer-of-the-biggest-house-in-washington/	130
https://www.washingtonpost.com/news/powerpost/wp/2018/01/12/jeff-bezos-donates-33-million-to-scholarship-fund-for-dreamers/	130
https://www.linkedin.com/pulse/jeff-bezos-banned-powerpoint-meetings-his-replacement-jason-cheong	130
https://www.linkedin.com/pulse/20130925133311-291225-amazon-ceo-jeff-bezos-had-his-top-execs-read-these-three-books	130
https://www.nytimes.com/2018/08/02/style/jeff-bezos-style-icon.html	130
https://www.nytimes.com/2018/09/21/us/bezos-montessori-preschool.html	130
https://www.nytimes.com/2018/05/09/opinion/jeff-bezos-spend-131-billion.html	130

URL	keyword	match	sentiment	magnitude
https://www.nytimes.com/2018/01/12/technology/jeff-bezos-amazon.html	Amazon	100	0.89999998	0.89999998
https://m.seattlepi.com/technology/businessinsider/article/AMAZON-CEO-I-Want-To-See-Millions-Of-People-5495986.php	Amazon	100	0.69999999	1.39999998
https://www.seattlepi.com/seattlenews/article/things-Jeff-Bezos-is-richer-than-13067498.php	Jeff Bezos	100	0.2	8.80000019
https://www.nytimes.com/2017/07/27/business/jeff-bezos-richest-man.html	Jeff Bezos	100	0.1	3.5999999
https://www.seattlepi.com/business/amp/Jeff-Bezos-is-planning-his-big-philanthropic-12997579.php	Amazon	100	0.1	10.6999998
https://www.seattlepi.com/local/amp/What-is-Jeff-Bezos-doing-12291658.php	Jeff Bezos	100	0.1	1.5
https://www.washingtonpost.com/education/2018/09/19/jeff-bezos-is-spending-billion-help-homeless-educate-poor-kids-sounds-good-is-it/	Jeff Bezos	100	0	2.5
https://www.washingtonpost.com/education/2018/09/19/jeff-bezos-is-spending-billion-help-homeless-educate-poor-kids-sounds-good-is-it/	Amazon	100	0	0
https://www.washingtonpost.com/education/2018/09/19/jeff-bezos-is-spending-billion-help-homeless-educate-poor-kids-sounds-good-is-it/	Amazon	100	0	0
https://www.washingtonpost.com/national/washington-post-to-be-sold-to-jeff-bezos/2013/08/05/ca537c9e-fe0c-11e2-9711-3708310f6f4d_story.html	Jeff Bezos	100	0	14.8000002
https://www.washingtonpost.com/national/washington-post-to-be-sold-to-jeff-bezos/2013/08/05/ca537c9e-fe0c-11e2-9711-3708310f6f4d_story.html	Amazon	100	0	5.4000001
https://www.washingtonpost.com/national/washington-post-to-be-sold-to-jeff-bezos/2013/08/05/ca537c9e-fe0c-11e2-9711-3708310f6f4d_story.html	Union	100	0	0
https://www.washingtonpost.com/business/economy/washington-post-closes-sale-to-amazon-founder-jeff-bezos/2013/10/01/fca3b16a-2acf-11e3-97a3-ff2758228523_story.html	Jeff Bezos	100	0	0.80000001
https://www.washingtonpost.com/business/economy/washington-post-closes-sale-to-amazon-founder-jeff-bezos/2013/10/01/fca3b16a-2acf-11e3-97a3-ff2758228523_story.html	Amazon	100	0	0.1
https://www.washingtonpost.com/national/jeff-bezos-on-post-purchase/2013/08/05/e5b293de-fe0d-11e2-9711-3708310f6f4d_story.html	Jeff Bezos	100	0	0
https://www.washingtonpost.com/national/jeff-bezos-on-post-purchase/2013/08/05/e5b293de-fe0d-11e2-9711-3708310f6f4d_story.html	Amazon	100	0	0
https://www.washingtonpost.com/news/reliable-source/wp/2017/01/12/jeff-bezos-is-the-anonymous-buyer-of-the-biggest-house-in-washington/	Jeff Bezos	100	0	0.60000002
https://www.washingtonpost.com/news/reliable-source/wp/2017/01/12/jeff-bezos-is-the-anonymous-buyer-of-the-biggest-house-in-washington/	Amazon	100	0	0.1
https://www.washingtonpost.com/news/powerpost/wp/2018/01/12/jeff-bezos-donates-33-million-to-scholarship-fund-for-dreamers/	Jeff Bezos	100	0	2.79999995
https://www.washingtonpost.com/news/powerpost/wp/2018/01/12/jeff-bezos-donates-33-million-to-scholarship-fund-for-dreamers/	Amazon	100	0	0.1
https://www.washingtonpost.com/lifestyle/style/jeffrey-bezos-washington-posts-next-owner-aims-for-a-new-golden-era-at-the-newspaper/2013/09/02/30c00b60-13f6-11e3-b182-1b3bb2eb474c_story.ht	Amazon	100	0	0.2
https://www.washingtonpost.com/news/comic-riffs/wp/2016/07/16/from-boyhood-to-hollywood-jeff-bezos-who-played-star-trek-as-a-kid-has-a-role-in-star-trek-beyond/	Jeff Bezos	100	0	4.0999999
https://www.linkedin.com/pulse/customer-support-done-right-jeff-bezos-e-mail-floors-indian-p-k	Jeff Bezos	100	-0.1	2
https://www.linkedin.com/pulse/customer-support-done-right-jeff-bezos-e-mail-floors-indian-p-k	Jeff Bezos	100	-0.1	0.2
https://www.linkedin.com/pulse/20141201155043-461078-jeff-bezos-a-profile-in-failure	Jeff Bezos	100	-0.1	12.3999996
https://www.seattlepi.com/business/amp/Jeff-Bezos-is-planning-his-big-philanthropic-12997579.php	Amazon	100	-0.1	0.1
https://www.thestranger.com/slog/2018/08/10/30563251/jeff-bezos-dreams-of-shooting-his-payload-into-space-while-we-all-pay-for-amazons-carbon-footprint	Jeff Bezos	100	-0.1	2.4000001
https://www.thestranger.com/slog/2018/07/20/29461096/why-dont-people-think-jeff-bezos-is-as-bonkers-as-imelda-marcos	Amazon	100	-0.2	0.2
https://www.linkedin.com/pulse/20141201155043-461078-jeff-bezos-a-profile-in-failure	Amazon	100	-0.3	0.89999998
https://www.nytimes.com/2018/01/12/technology/jeff-bezos-jumps-into-dreamer-fight-with-gift-for-scholarships.html	Amazon	100	-0.3	1

POTENTIAL USES

- Monitor the online sentiment about product, person, event.
- Monitor for potential data leaks containing sensitive keywords.
- Automate portions of on-going OSINT gathering.

TODO

- Implement selenium to add support for logging into websites (social media, news sites, etc)
- Tracking results across time to monitor the changes in sentiment automatically.
- Community contributions?

QUESTIONS?